Sonatype Nexus vs GitLab

GitLab compared to other DevOps tools

Sonatype Nexus Platform is comprised of multiple products which contribute to the Sonatype Nexus security capabilities. Those products are Nexus Lifecycle, Nexus Auditor, Nexus Firewall, and Nexus Repository Pro, and the Nexus Intelligence service. In the application security space, Sonatype Nexus scans open source components for security vulnerabilities, scans containers and offers license management.

GitLab Ultimate automatically includes broad security scanning with every code commit including Static and Dynamic Application Security Testing, along with dependency scanning, container scanning, and license management.

For packaging deployments, both Sonatype and GitLab offer container registery, but Sonatype also offers a full binary repository in the form of Nexus Repository (available in both OSS and Pro).

FEATURES	Sonatype Nexus	GitLab
Built-in Container Registry GitLab Container Registry is a secure and private registry for Docker images. It allows for easy upload and download of images from GitLab CI. It is fully integrated with Git repository management. Documentation on Container Registry	✓	✓
Full Binary Repository A binary repository is a software repository for packages, artifacts and their corresponding metadata. It can be used to store binary files produced by an organization itself, such as product releases and nightly product builds, or for third party binaries which must be treated differently for both technical and legal reasons.	✓	✗
Static Application Security Testing GitLab allows easily running Static Application Security Testing (SAST) in CI/CD pipelines; checking for vulnerable source code or well known security bugs in the libraries that are included by the application. Results are then shown in the Merge Request and in the Pipeline view. This feature is available as part of [Auto DevOps](https://docs.gitlab.com/ee/topics/autodevops/#auto-sast) to provide security-by-default. Learn more about Static Application Security Testing	✗	✓

Dependency Scanning

GitLab automatically detects well known security bugs in the libraries that are included by the application, protecting your application from vulnerabilities that affect dependencies that are used dynamically. Results are then shown in the Merge Request and in the Pipeline view. This feature is available as part of [Auto DevOps](https://docs.gitlab.com/ee/topics/autodevops/#auto-dependency-scanning) to provide security-by-default.

Learn more about Dependency Scanning

 

Container Scanning

When building a Docker image for your application, GitLab can run a security scan to ensure it does not have any known vulnerability in the environment where your code is shipped. Results are then shown in the Merge Request and in the Pipeline view. This feature is available as part of [Auto DevOps](https://docs.gitlab.com/ee/topics/autodevops/#auto-container-scanning) to provide security-by-default.

Learn more about container scanning

 

Dynamic Application Security Testing

Once your application is online, GitLab allows running Dynamic Application Security Testing (DAST) in CI/CD pipelines; your application will be scanned to ensure threats like XSS or broken authentication flaws are not affecting it. Results are then shown in the Merge Request and in the Pipeline view. This feature is available as part of [Auto DevOps](https://docs.gitlab.com/ee/topics/autodevops/#auto-sast) to provide security-by-default.

Learn more about application security for containers

 

Interactive Application Security Testing

[IAST](https://blogs.gartner.com/neil_macdonald/2012/01/30/interactive-application-security-testing/) combines elements of static and dynamic application security testing methods to improve the overall quality of the results. IAST typically uses an agent to instrument the application to monitor library calls and more. GitLab does not yet offer this feature.

 

Runtime Application Security Testing

RASP uses an agent to instrument the application to monitor library calls as the application is running in production. Unlike other security tools, RASP can take action to block threats in real-time, similar to a Web Application Firewall but from within the app's runtime environment rather than at the network layer. GitLab does not yet offer this feature.

 

License Management

Check that licenses of your dependencies are compatible with your application, and approve or blacklist them. Results are then shown in the Merge Request and in the Pipeline view.

Learn more about License Management

Maven Repository

GitLab's Maven repository makes it easier to publish and share Java libraries across an organization, and ensure dependencies are managed correctly. It is fully integrated with GitLab, including authentication and authorization.

Documentation on the Maven Repository